Project 3

CSE 473/573 (Spring 2019)

Due Date: May 13th, 11:59 pm

1 Face Detection in the Wild [100]

The goal of this project is to have you implement the Viola-Jones face detection algorithm [1]. You should download the paper and understand its details. As discussed in class, the Viola-Jones algorithm is capable of detecting frontal faces in real time and is regarded as a milestone in the development of computer vision. Despite the fact that deep learning-based face detectors [2, 3] have gradually emerged as the preferred paradigm for face detection, variants of Viola-Jones algorithm still remain competitive in situations where speed is critical. A great introduction to Viola-Jones algorithm can be found at `https://www.youtube.com/watch?v=uEJ71VlUmMQ`.

watched

2 Project Description

Given a face detection dataset composed of thousands of images, the goal is to train a face detector using the images in the dataset. The trained detector should be able to locate all the faces in any image coming from the same distribution as the images in the dataset. Figure 1 shows an example of performing face detection. We will use FDDB [4] as the dataset for this project. FDDB contains more than 2800 images and associated bounding box annotation, for more than 5700 faces.

Please keep in mind that Viola-Jones algorithm [1] is the ONLY method that you could use for this project. Using other methods - HOG / SIFT / SURF + SVM, deep learning-based methods, will result in a deduction of more than 50% of the maximum possible points for this project.

You must also use integral images to implement the feature extraction. You are free to use enhancements of the boosting algorithm such as cascades to obtain better results.

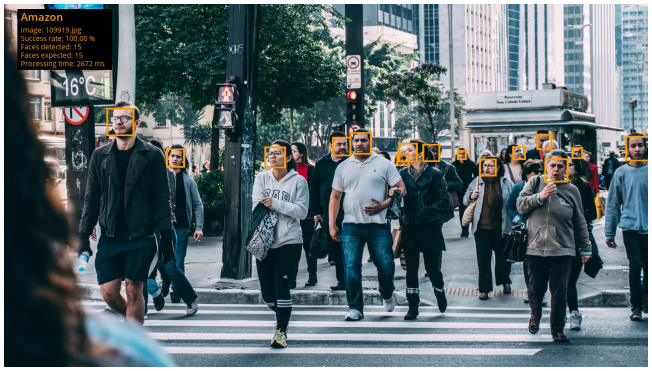

Figure 1: An example of performing face detection. The detected faces are annotated using orange bounding boxes.

3 Libraries permitted and prohibited

- Any Python Standard Library can be used.
- Any APIs provided by OpenCV that have "`cascade`", "`Cascade`", "`haar`" or "`Haar`" functionality can not be used.
- Using any APIs that implement part of Viola-Jones algorithm directly, *e.g.*, an API that computes integral image, will result in a deduction of 10% – 100% of the maximum possible points of this project[1].

[1]Please do not ask if a specific API could be used on Piazza. You need to judge whether an API implements part of Viola-Jones

4 Data and Evaluation

The FDDB [4] and the evaluation script should be downloaded direction from the following site: http://vis-www.cs.umass.edu/fddb/.

The following bash script will be used to evaluate the performance of the face detector you train:

```
unzip UBID-project3.zip
cd [UBID-project3]
python3 YourFaceDetector.py [data-directory]
python3 ComputeFBeta.py ./results.json [ground-truth.json]
```

`YourFaceDetector.py` contains YOUR python code that will able to detect faces in all the images in `[data-directory]` and generate a json file, *i.e.*, `results.json`, that stores all the bounding boxes of the detected faces. The bounding boxes of the detected faces should be stored in a list using the following format:

```
[{"iname": "img.jpg", "bbox": [x, y, width, height]}, ...]
```

`"img.jpg"` is an example of the name of an image; `x` and `y` are the row index and column index of the top-left corner of the bounding box; `width` and `height` are the width and height of the bounding box, respectively. `x`, `y`, `width` and `height` should be integers.

`ComputeFBeta.py` contains OUR python code that computes f_β using `results.json` and the ground truth in `[ground-truth.json]`. This will be uploaded by the end of the day on 4/15/19.

5 Code and Report

Your code should be zipped up in a directory called Project3. Since this project allows more flexibility in the use of libraries, please also remember to include a file with your resources your code requires to make it easier on the grader. This should be in the form of a txt file named "requirements.txt". The file named "requirements.txt" should specify the libraries you used and the version of the libraries, and our grader Liyuan should be able to install all the libraries you used by using the command: pip install requirements.txt

In addition to your code, a report is required with this project. It should contain

- Your name
- An overview of the Viola-Jones algorithm,
- A description of your implementation,
- Results of your face detector on FDDB (from the program above) and
- An analysis of the results (failure cases, possible improvements, etc),

The report should be a pdf file. You will be graded on both the report and an evaluation on a held out test dataset that you currently do not have access to

6 Academic Integrity

There are more than a dozen open source python implementations of Viola-Jones algorithm, including but not limited to those listed on https://github.com/topics/viola-jones?l=python. Copying and pasting code from open source implementations is strictly prohibited. For details about academic integrity policy of UB, please refer to https://catalog.buffalo.edu/policies/integrity.html.

References

[1] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'01)*, 2001.

[2] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *IEEE International Conference on Computer Vision (ICCV'15)*, December 2015.

[3] Y. Sun, X. Wang, and X. Tang, "Deeply learned face representations are sparse, selective, and robust," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'15)*, June 2015.

[4] V. Jain and E. L. Miller, "Fddb: a benchmark for face detection in unconstrained settings," 2010.

algorithm directly on your own.